## AMENDMENT TO THE SPECIFICATION

Please replace the paragraph starting at page 7, line 7 with the following:

Fig. 10 is ~~a flow diagram~~ an illustration of a circuit for generating a compressed pseudo bit fail map according to an embodiment of the present invention;

Please replace the paragraph starting at page 10, line 11 with the following:

The pseudo bit map generation involves the repetition of the same test pattern, while generating pass/fail information for a limited address space within the array. Different address limitations and test patterns generate fail strings having different scopes, and thus can be tailored to a specific application.

Please replace the paragraph starting at page 10, line 15 with the following:

A memory device has x x-addresses, including bank addresses, and y y-addresses is able to address $2^{(x+y)}$ memory cells. According to one embodiment of the present invention, only a part of a memory array is tested by limiting the data strobe at read operations, where one or more selected addresses are fixed at 0 or 1. Since the full array is to be tested, the pattern is duplicated for all address combinations. Fig. 2 shows all address combinations for a 4x4 device with two x addresses (X0 and X1) and two y addresses (Y0 and Y1) and eight test steps including a corresponding pass/fail string 200. The pass/fail string 200 shows, for every test, a pass (.) or fail (F). For address X0 in the first combination 204, X0=0 is shown as 1 labeled with ~~find~~ numeral 201 and X0=1 is shown as 2 labeled with ~~find~~ numeral 202. When each combination 204 to 207 is superimposed upon one-another, all of the possible address combinations are shown for the 4x4 device. Further, by assigning a pass or a fail to each number, one through 8,

according to the pass/fail string 200, and applying ~~and~~ an AND function to the superimposed combinations, e.g., 1,3,5,7 for the upper left of each combination, a bit fail map can be generated. For example, the superimposed combination 1(.), 3(F), 5(.), 7(.) yields a pass according to the Boolean operation AND for the bit corresponding to that combination.

Please replace the paragraph starting at page 11, line 30 with the following:

To identify all given fail types the minimum number of x addresses and y addresses are listed: X0, X1, X8, X9, X10, X11, X12, Y0, Y1, Y9, Y10, Y11. These twelve addresses correspond to 4096 cells ($2^{12} = 4096$), as shown in Fig. 3.

Please replace the paragraph starting at page 15, line 11 with the following:

The fail string generates for X0, where '0' is passing and '1' is failing, for the fail address X0=1. Since a WL has more than one cell failing certain addresses, for example, Y0, will fail regardless, whether they are held high or low. In the address this can be expressed by a '-'. A short form of the failing addresses can be given by (y,x) =(0----,0110111). For every '-' 2 fail address are generated resulting in a total of 2*2*2*2 = 16 failing cells in a pseudo bit fail map. The full bit fail map with these addresses would be ~~212~~ $2^{12}$ = 4096 pixels large. For visualization, it is better to partition the picture into 2 maps (see Fig. 5), a first map 501 for the details using x0, x1, y0, y1 and a second map 502 for the global overview using x8, x9, x10, x11, x12 and Y9, Y10, Y11.